

CSC110 Project Proposal: Country-wise Financial Responsibility for Climate Change

Ansh Jain, Nikhil Sreekumar, Sidharth Sachdev

Monday, December 14, 2020

Problem Description and Research Question

Change has been exponentially quick in the last century, and it certainly has not been kind to the climate of our world. Instead of guiding us away from the brink of destruction, many of our leaders have steered us right to the tipping point. Our problem is that some very privileged people are allowed to make decisions that adversely affect the entire planet. We have decided to research how various factors contribute to the individual impact of a nation on the worsening climate of the earth. We want to quantitatively find out how much each country is responsible for climate change.

Hence our research question is,

What is each country's responsibility, in monetary value, towards confronting the looming climate emergency based on certain factors, ranging from the country's effect on the earth's climate to its fiscal ability?

The world runs on money, and any form of commitment on a global front must start off with a budget. This resulted in us choosing a monetary approach to answer the research question. However, the aim of the project is not realized through calculating this aforementioned budget, a role imposed upon the user, but is done so by producing a fair distribution of this budget amongst the nations of the world.

We intend to go about doing so by creating a desktop application that allows the user to gain this information in whichever way is more preferable: comparing nations in a choropleth map or analyzing each country separately. It is also imperative that the application has a clean and understandable User Interface so that the user is aware of the expected input as well as the features that come with it.

A major feature of the application is providing the user with the freedom to manipulate the datasets, weightage, and factors while using a set of standard underlying equations. There is a default directory with datasets that the users can use, but they are encouraged to delete or replace them with datasets of their own before running the program. The weight of each factor, the budget, and the year of analysis can be directly altered while running the application. Together, this customizability allows for an interactive application that can produce several hypothetical budget distributions in a systematic, unbiased manner

Dataset Description

There are two directories in which datasets are stored. 'Responsibility Datasets' is where the user enters datasets as per what best fits their idea of assigning responsibility to each nation state. 'Constant Datasets' cannot be altered by the user since they are standard values in the computational formula.

CONSTANT DATASETS

1. GDP is a dataset of the CSV format which provides basic information about each country such as GDP in from 1960 to 2017. The data has been compiled by World Bank, and OECD National Accounts.
2. countries_codes_and_coordinates contain information regarding location and ISO 2 and 3 codes. The country name and ISO code are used. Dataset obtained from a Github repository under username Tadast.

RESPONSIBILITY DATASETS (DEFAULT)

1. CO2 emission is a dataset of the CSV format which provides information on the emissions of Carbon Dioxide in metric tons based on each country from 1990 to 2016. This dataset has been sourced from data.world and was created by Oak Ridge National Laboratory.
2. Percentage use of renewable energy is a dataset of the CSV format which provides information on the percentage of renewable energy used out of the total energy consumption per country from 1960 to 2015. The dataset has been sourced from the World Bank.
3. CRI is a dataset of the CSV format which provides information on categories such as score, fatalities, losses in millions of dollars based on the effects of climate in each country in 2014. The source of the data set is GERMANWATCH.
4. The same GDP dataset from Constant Datasets can also serve as a factor. Thus, it is also a part of the Responsibility Datasets

Note: Each Dataset provided has data for a range of years. However, the data for CRI is from the year 2014 exclusively. Hence, the only common year available to the user for analysis is 2014. The user can change this by editing the directory containing datasets as required. The user's preference and available data may differ and they have the freedom to input datasets in their best interest. This is elaborated further in the Computational Plan.

Computation

`dataset_utilities.py` is a script that is responsible for bridging the GUI aspect of the application with the computation. `_extract_wanted_column` extracts the required columns from the datasets provided, and using `name_to_iso`, provides a raw amalgamation of input csv files into a mapping in `raw_datasets`. These functions are the building blocks to the following public functions that are used by other scripts.

1. `map_iso_to_country` creates a key-value pair of ISO code to Country dataclass. It removes all insufficient data from which a weighted average cannot be calculated. It is used to find all the countries that are to be displayed in the map.
2. `get_clean_datasets` converts `map_iso_to_country` back into the form of `get_raw_datasets`. Is important for all country related computation in `computation.py`
3. `possible_years` Uses the information from `get_clean_datasets` to return which years useful data can be derived from.

The main GUI script, `user_input.py` contains two windows in the application are built in an Object Oriented implementation, both inheriting the framework provided by `cclm_application_window.py`

The application first takes in user input for a year to Analyze. The user is provided a drop-down of possible years to choose from.

If the following preconditions:

1. $budget \geq \$1,000,000$
2. Sum of all constants for weighted average = 100

Are not satisfied, an error dialog is created.

As the application is executed, the `map_win_open` method in `user_input.py` calls the `run` method in the script relating to map plotting, `map.py`. Method `run` is a public method that displays a choropleth map giving information about the budget, budget percentage, name of the country, its iso-code and each country's data for every factor. This data can be viewed by hovering over the country on the map using the package pandas' `DataFrame` method to pass data into a choropleth object of `plotly.express` module that shows the map. The budget and budget percentage for each country are obtained by calling the private `_plot` method that takes in the required information and passes over the values for computation in `computation.py`

`_plot` calls `budget_details` method in `computation.py` to return a dictionary mapping each country code to a corresponding tuple of the budget and budget percentage.

`budget_details` is a public method that calculates the appropriate amount in US dollars to be allocated to each country and returns a tuple containing both budget and budget percentage for that specific country based on the formula,

$$Budget(A) = Responsibility(A) \cdot Total_Budget$$

$$GDP\% = \frac{Budget(A)}{Country's\ GDP} \cdot 100$$

It does this by calling the private method `_responsibility`. It carries out the major chunk of the calculation by first checking whether there is any unavailable data for a country in a specific factor (this is only if the country has data for at least one of the factors). If this is the case, then the country is removed from the dict and not displayed on the map.

`_responsibility` also calls private methods `_negative_calculation` and `_positive_calculation` for each factor based on their relation of ‘direct’ or ‘inverse’ as input by the user.

- if the word ‘direct’ is used, it means that the factor is positively correlated to climate change like GDP, Carbon Dioxide Emissions, and Climate Risk Index. It uses the formula,

$$\frac{Country's\ data}{TotalData}$$

- if the word ‘inverse’ is used, it means that the factor is negatively correlated to climate change like Renewable Energy.

$$\frac{Total\ data - Country's\ data}{\sum_{i \in Country\ data} Total\ data - i}$$

The result from these calculations(point out of 1) are then multiplied with the corresponding weight of the factor. This is done for each factor and their sum(out of 100) gives the responsibility of that country. Assuming there are 4 factors (as is in the default):

$$Responsibility = \alpha F1 + \beta F2 + \gamma F3 + \delta F4$$

A way to crosscheck these values is if the responsibilities of all countries are added, we get 100%. Hence we know that the computation is right.

Using these methods, we were able to come up with an implementation for our plan and display the choropleth map along with the information of each country when the user hovers over that region.

References

1. “Qt Documentation.” Qt for Python - Qt for Python, doc.qt.io/qtforpython/.
2. “Choropleth Maps.” Plotly, plotly.com/python/choropleth-maps/.
3. Oak Ridge National Laboratory. CO2 Emissions (Kt), data.worldbank.org/indicator/EN.ATM.CO2E.KT.
4. “Renewable Energy Consumption (% of Total Final Energy Consumption).” Data, data.worldbank.org/indicator/EG.FEC.RNEW.ZS.
5. Eckstein, David. “Global Climate Risk Index 2014.” Germanwatch.org, <https://germanwatch.org/en/7659>.
6. World Bank, and OECD National Accounts. “GDP (Current US\$).” Data, n.d. <https://data.worldbank.org/indicator/NY.GDP.MKTP.CD>.
7. Tamošauskas, Tadas. “Countries with Their (ISO 3166-1) Alpha-2 Code, Alpha-3 Code, UN M49, Average Latitude and Longitude Coordinates.” Gist. Accessed December 14, 2020. <https://gist.github.com/tadast/8827699>.